

## TECHNICAL MEMORANDUM (NASA) 12

## OMEGA FLIGHT-TEST DATA REDUCTION SEQUENCE

A series of FORTRAN computer programs for preparation and summary of flight-test data obtained from the Ohio University Omega Receiver.

by

Robert W. Lilley  
Avionics Engineering Center  
Department of Electrical Engineering  
Ohio University  
Athens, Ohio 45701

November, 1974

Supported by

National Aeronautics and Space Administration  
Langley Research Center  
Langley Field, Virginia  
Grant NGR 36-009-017



(NASA-CR-140827) OMEGA FLIGHT-TEST DATA  
REDUCTION SEQUENCE (Ohio Univ.) 15 p HC  
\$3.25 CSCL 17G

N75-11911

Unclass

63/04 02422

## TABLE OF CONTENTS

	Page
I. INTRODUCTION	1
II. FORTRAN COMPUTER PROGRAMS	2
A. Flight Data Convert - FDCON	2
B. Flight Data Summary - FDSUM	5
C. Data Copy - DACOP	10
III. SUMMARY	12
IV. ACKNOWLEDGEMENTS	13

## I. INTRODUCTION

This technical memorandum presents three computer programs used at Ohio University for Omega data conversion, summary, and preparation for distribution. Program logic and sample data formats are included, along with operational instructions for each program.

Flight data (or data collected in flight format in the laboratory) is provided by the Ohio University Omega receiver base in the form of 6-bit binary words representing the phase of an Omega station with respect to the receiver's local clock. All eight Omega stations are measured in each 10-second Omega time frame. In addition, an event-marker bit and a time-slot-D synchronizing bit are recorded. Program FDCON is used to remove data from the flight recorder tape and place it on data-processing cards for later use.

Program FDSUM provides for computer plotting of selected LOP's, for single-station phase plots, and for printout of basic signal statistics for each Omega channel. Mean phase and standard deviation are printed, along with data from which a phase distribution can be plotted for each Omega station.

Program DACOP simply copies the Omega data deck a controlled number of times, for distribution to users.

These programs were developed on the Ohio University Computer Services' System/360 Model 44 computer. Any special coding which may be system-dependent is noted in the individual program descriptions.

## II. FORTRAN COMPUTER PROGRAMS

The computer routines reported here were developed on the Ohio University IBM System/360 Model 44 computer, using standard FORTRAN IV language. Two routines are used by program FDCON which are system-dependent. These routines are fully described so that non-Ohio University users may write suitable routines for use at their installations.

A. Flight Data Convert - FDCON. The listing for FDCON appears on the following pages. The purpose of this routine is to read the Omega data-collection tape (produced by the Kennedy 1600 incremental tape recorder) and produce punched cards containing the commutated phase data from the eight Omega stations. The Kennedy tape is generally produced as a one-record file containing 65,535 bytes or less, followed by a tape mark (end-of-file) indicator.

FDCON punches cards with 24 two-digit Omega phase measurements each, followed by the eight-digit card sequence number. Each phase measurement consists of two digits, with the number in the range 00-63. If a minus sign appears in the three-digit field reserved for each measurement, it indicates an event mark placed at this position in the data. Event marks should be separately documented for each data set.

On the last card of a deck, the end of phase data is signaled by a 99 in the next measurement position.

Card deck data always begins with Omega station D, referenced to the 10.2 KHz time slot. Each card contains three complete Omega sequences of eight measurements each.

FDCON uses two system-dependent routines which are generally written in the particular machine language being used. Subroutine TAPERR is called in the Ohio University system to set the number of tape error retries which will be attempted by the system. Since Kennedy digital tapes have, from our experience, shown a relatively high error rate when reading very long records, the retry rate is set relatively low.

Subroutine VTAP2 allows the recovery of variable-length records regardless of tape errors. See the FORTRAN listing at statement number 15. The routine fills the buffer IN from data set 4 until either the end-of-record signal or 65,535 bytes are transferred, whichever occurs first. The actual number of bytes transferred to storage is returned in IACTL. ERR is a 3-word error buffer, not used in this program. Statement number 53 is taken on end-of-file or tape mark signal, and statement 23 is taken on tape errors.

VTAP2 transfers data until record end or an error occurs. In FDCON, we wish to recover all possible data, so data transferred previous to the tape error is available for conversion to cards.

A printed listing of output data is produced, showing minus signs for event marks, along with phase data and card sequence numbers as integers. The program places

C OMEGA TAPE CONVERT  
C FDCON

C READS OMEGA DIGITAL TAPE (65535 BYTES MAX) TO TAPE MARK AND CONVERTS  
C TO CARDS IN I3 FORMAT. EVENT MARKS RESULT IN NEGATIVE NUMBERS.

C READS UNTIL IT HITS 2 CONCURRENT TAPE MARKS, OR, NUMBER OF FILES  
C GIVEN ON INPUT CONTROL CARD: FORMAT: FILES=N, WHERE N=1-9

C R. W. LILLEY, AVIONICS, NOVEMBER, 1974

C INTEGER E,BK  
C DATA E/'E'//,BK/' '  
C INTEGER ERR(3),CBUF(24)/24\*99/  
C LOGICAL\*1 IN(65535)/65535\*.FALSE./  
C LOGICAL\*1 IA(4)  
C EQUIVALENCE(ITMP,IA(1))

C...SET NUMBER OF ERROR RETRIES FOR TAPE

CALL TAPERR(2)

IPCH=2

ITMP=0

IFILE=99

READ(1,1,END=10)IFILE

1 FORMAT(6X,I1)

10 ISW=0

DO 20 I=1,IFILE

IC=0

ITS=0

IBF=0

C...USE OHIO UNIV. MACHINE-LANGUAGE TAPE READ TO GET AROUND

C...ERROR PROCESSING, ETC. (SEE WRITEUP)

15 CALL VTAP2(4,IN,65535,IACTL,ERR,653,623)

724 IF(IACTL.LT.0)STOP 1

PRINT 24,I,IACTL

24 FORMAT(' RECORD IN FILE ',I1,' : ',I6,' BYTES')/)

ISW=0

ISW2=1

DO 28 J=1,IACTL

IA(4)=IN(J)

ITS=ITS+1

IF(ITS.EQ.9)ITS=1

IE=0

C...CHECK FOR EVENT

IQ=ITMP-128

C ERROR - OUT OF SYNC.

LIN=BK

IF(IQ.GE.64.AND.ITS.EQ.1)LIN=E

IF(LIN.EQ.E)IPCH=0

IF(IQ.LT.0) IE=1

IF(IQ.LT.0) IQ=IQ+128

C...CHECK FOR SYNC.

IF(IQ.GE.64)IQ=IQ-64

IF(IE.EQ.1) IQ=-IQ

IBF=IBF+1

IF(IBF.GT.24)GO TO 120

130 CBUF(IBF)=IQ

GO TO 28

```

C PUNCH
120 IC=IC+1
    WRITE(IPCH,121)CBUF,IC
121 FORMAT(24I3,I8)
    PRINT 122,CBUF,IC,LIN
122 FORMAT(10X,24I3,I8,5X,A1)
    DO 123 K=1,24
123 CBUF(K)=99
    IBF=1
    GO TO(130,20,155),ISW2
28 CONTINUE
    GO TO 15
--C EDF
53 PRINT 54
54 FORMAT(' TAPE MARK'//)
    IF(ISW.EQ.1)STOP
    ISW=1
    ISW2=2
    IF(IBF.EQ.24)ISW2=3
    GO TO 120
155 ISW2=2
    GO TO 120
20 CONTINUE
    STOP
23 PRINT 723
723 FORMAT(' ***TAPE ERROR'//)
    GO TO 724
    END

```



```

C...OMEGA-TEST DATA SUMMARY PROGRAM - FDSUM
C...READS OMEGA DATA DECK FROM FDCON AND PRODUCES PLOTS OF SELECTED
C...PHASE MEASURES, SUMS OR DIFFERENCES; PRINTS MEAN, S. D., AND
C...SPECTRAL DATA FOR EACH OMEGA TIME SLOT.
C...
C...INPUT DECK FROM FDCON IS PRECEDED BY ONE CONTROL CARD. FORMAT:
C...
C... A-B,D+G - GIVES TWO PLOTS ONE FOR DIFFERENCE OF A AND B PHASE,
C... AND ONE FOR SUM OF D AND G PHASE.
C...ONE PLOT MAY BE DONE BY ELIMINATING COMMA AND SECOND EXPRESSION
C...ONE-STATION PHASE PLOT MAY BE MADE BY USING THE DESIRED STATION
C... LETTER FOLLOWED BY TWO BLANKS.
C... BLANK CARD YIELDS NO PLOT AT ALL
C...
C... R. W. LILLEY, AVIONICS, NOVEMBER, 1974
C
      INTEGER SBUF(64,8)/512*0/
      INTEGER BK/' ',STAR/'*'/,E/'E'/,LINE(132),LQ(9)
      DIMENSION IN(24),EN(8),SUM(8),SUMSQ(8),EM(8),ESD(8)
      INTEGER IBAR/'|'/
      INTEGER KL(4)
      INTEGER IP/'+'/,LA(3),LB(3),LT(9)/'D','E','F','G','H','A','B',
      * 'C',' ' /
      IPS=1
      LQ(9)=0
      DO 20 I=1,8
      EN(I)=0.
      SUM(I)=0
20    SUMSQ(I)=0.
C...READ CONTROL CARD AND PROCESS IT
      READ(1,2)LA,LB
      2 FORMAT(3A1,1X,3A1)
      DO 26 I=1,4
      KL(I)=0
26    DO 16 I=1,9
      IF(LA(1).EQ.LT(I))KL(I)=I
      IF(LA(3).EQ.LT(I))KL(2)=I
      IF(LB(1).EQ.LT(I))KL(3)=I
      IF(LB(3).EQ.LT(I))KL(4)=I
16    CONTINUE
      DO 128 I=1,4
      IF(KL(I).NE.9)GO TO 129
128    CONTINUE
      IPS=2
129    DO 17 I=1,4
      IF(KL(I).GT.0)GO TO 17
      PRINT 18
18    FORMAT(' *** CONTROL CARD ERROR')
      STOP
17    CONTINUE
      LAM=1
      IF(LA(2).EQ.IP)LAM=-1
      LBM=1
      IF(LB(2).EQ.IP)LBM=-1
      IF(IPS.EQ.2)GO TO 10
      PRINT 9,LA,LB
9    EFORMAT('10 OMEGA PLOTS: ',3A1,64X,3A1/' 10 SECONDS PER LINE (DOWN',
      * ')'; 1/64 LANE PER CHARACTER (ACROSS).'/)
      PRINT 786
786    FORMAT('/' 0',T65,'63',T68,'0',T130,'63')
      PRINT 787
787    FORMAT(' ',130(' _'))
C...READ INPUT DECK
10    READ(1,1,END=50)IN
1    FORMAT(24I3)
C...PROCESS EACH PHASE MEASUREMENT ON CARD
      DO 11 J=1,3
C...CLEAR PLOT BUFFER
      DO 13 I=1,132
13    LINE(I)=BK
      LINE(3)=IBAR
      LINE(68)=IBAR
      LINE(66)=IBAR
      LINE(131)=IBAR
      DO 12 K=1,8
      LL=(IN((J-1)*8+K))

```



```

C...CHECK FOR END OF DATA (99)
  IF(LL.GF.99)GO TO 50
  L=IABS(LL)
  SUM(K)=SUM(K)+L
  EN(K)=EN(K)+1.
  SUMSQ(K)=SUMSQ(K)+L*L
  LQ(K)=L
  SBUF(L+1,K)=SBUF(L+1,K)+1
  IF(LL.LT.0) LINE(2)=E
12  CONTINUE
C PLOT
  IF(IPS.EQ.2)GO TO 10
  IF(KL(1).EQ.9.AND.KL(2).EQ.9)GO TO 456
  LOP1=LQ(KL(1))-LQ(KL(2))*LAM
  IF(LOP1.LT.0)LOP1=64+LOP1
  IF(LOP1.GE.64)LOP1=LOP1-64
  LINE(LOP1+3)=STAR
456 IF(KL(3).EQ.9.AND.KL(4).EQ.9)GO TO 457
  LOP2=LQ(KL(3))-LQ(KL(4))*LBM
  IF(LOP2.LT.0)LOP2=64+LOP2
  IF(LOP2.GE.64)LOP2=LOP2-64
  LINE(LOP2+68)=STAR
457 CONTINUE
  PRINT 112,LINE
112 FORMAT(132A1)
11  CONTINUE
  GO TO 10
C PRINT STIX
50 PRINT 101
101 FORMAT('1 STATISTICS FOR EACH TIME SLOT'//)
200 FORMAT(10X'D   E   F   G   H   A   B   C'//)
  PRINT 200
  DO 102 I=1,8
  EM(I)=SUM(I)/EN(I)
102 ESD(I)=SQRT ((SUMSQ(I)-((SUM(I)**2)/EN(I)))/EN(I))
  PRINT 103,EM,ESD
103 FORMAT(' MEAN   ',8(F4.1,1X))// ' S.D.   ',8(F4.1,1X)//
  PRINT 104
104 FORMAT(' SPECTRUM DATA'//)
  PRINT 200
  DO 105 I=1,64
  K=I-1
  PRINT 106,K,(SBUF(I,L),L=1,8)
106 FORMAT(1X,I2,5X,8(14,1X))
105 CONTINUE
  STOP
  END

```

# STATISTICS FOR EACH TIME SLOT

	D	E	F	G	H	A	B	C
MEAN	31.8	29.7	31.5	31.7	31.4	31.3	32.6	30.5
S.D.	18.3	18.4	18.6	18.4	18.4	18.5	17.5	17.8

## SPECTRUM DATA

	D	E	F	G	H	A	B	C
0	13	17	13	12	14	11	11	12
1	13	17	12	18	15	10	9	11
2	16	12	23	14	12	20	13	13
3	12	15	18	14	17	21	14	14
4	13	17	15	7	13	19	13	12
5	14	15	11	12	16	12	14	16
6	20	16	14	21	13	9	13	9
7	12	14	12	17	19	19	19	12
8	9	16	9	13	11	14	8	13
9	15	15	13	14	9	12	5	14
10	13	13	13	15	16	13	8	12
11	11	19	14	8	11	16	14	24
12	10	17	13	18	13	14	13	25
13	17	11	13	10	18	12	15	18
14	9	18	13	13	20	11	12	14
15	14	13	14	9	8	14	8	18
16	12	16	18	16	14	10	9	11
17	18	20	17	12	9	16	12	15
18	10	14	11	15	17	17	19	13
19	15	18	11	17	12	19	15	12
20	15	16	20	20	15	10	11	15
21	20	34	14	18	22	14	7	11
22	13	26	11	12	9	20	11	15
23	21	18	21	7	13	9	17	24
24	13	8	17	18	14	12	20	19
25	12	14	13	11	15	16	15	22
26	13	12	12	6	16	18	8	12
27	16	15	13	14	18	18	13	13
28	13	9	15	17	15	10	17	17
29	14	9	9	20	10	10	13	14
30	15	8	16	11	12	17	20	15
31	19	9	13	12	14	16	17	14
32	12	14	15	9	10	10	20	21
33	15	11	17	19	20	16	15	16
34	13	9	8	17	13	16	18	9
35	13	13	12	14	15	13	20	15
36	15	18	14	18	17	7	19	14
37	11	17	20	16	17	16	18	13
38	17	12	16	11	20	13	26	12
39	14	13	13	19	12	17	13	19
40	16	13	10	14	12	13	17	12
41	15	10	10	20	13	12	10	11
42	16	13	15	15	12	16	17	13
43	8	15	17	11	18	22	16	19
44	17	16	10	15	17	11	12	8
45	12	15	23	16	8	13	13	15
46	18	8	13	10	11	7	15	12
47	14	8	11	10	12	17	10	17
48	14	8	13	9	10	18	17	15
49	13	13	19	14	20	9	14	13
50	16	15	7	20	15	13	17	12
51	14	19	13	16	13	17	19	11
52	13	17	17	18	17	16	19	13
53	14	20	22	9	15	11	19	12
54	19	13	9	18	10	9	19	17
55	11	9	15	10	16	18	19	6
56	14	14	15	13	13	20	13	16
57	11	10	11	10	16	5	8	9
58	17	11	12	20	10	19	12	14
59	16	9	18	16	14	13	8	11
60	13	17	15	16	10	15	5	13
61	17	15	15	10	14	12	18	11
62	14	8	12	9	14	15	10	11
63	11	9	15	20	19	14	13	13

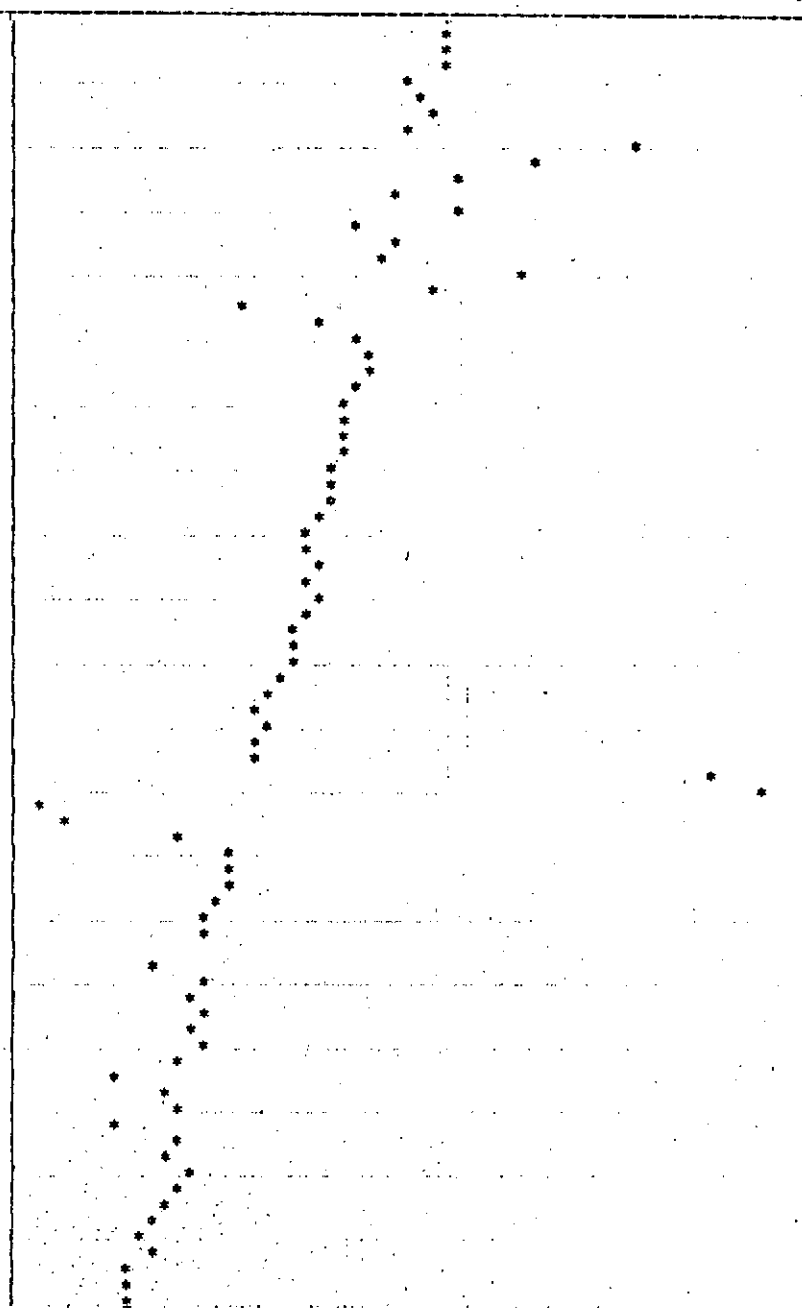
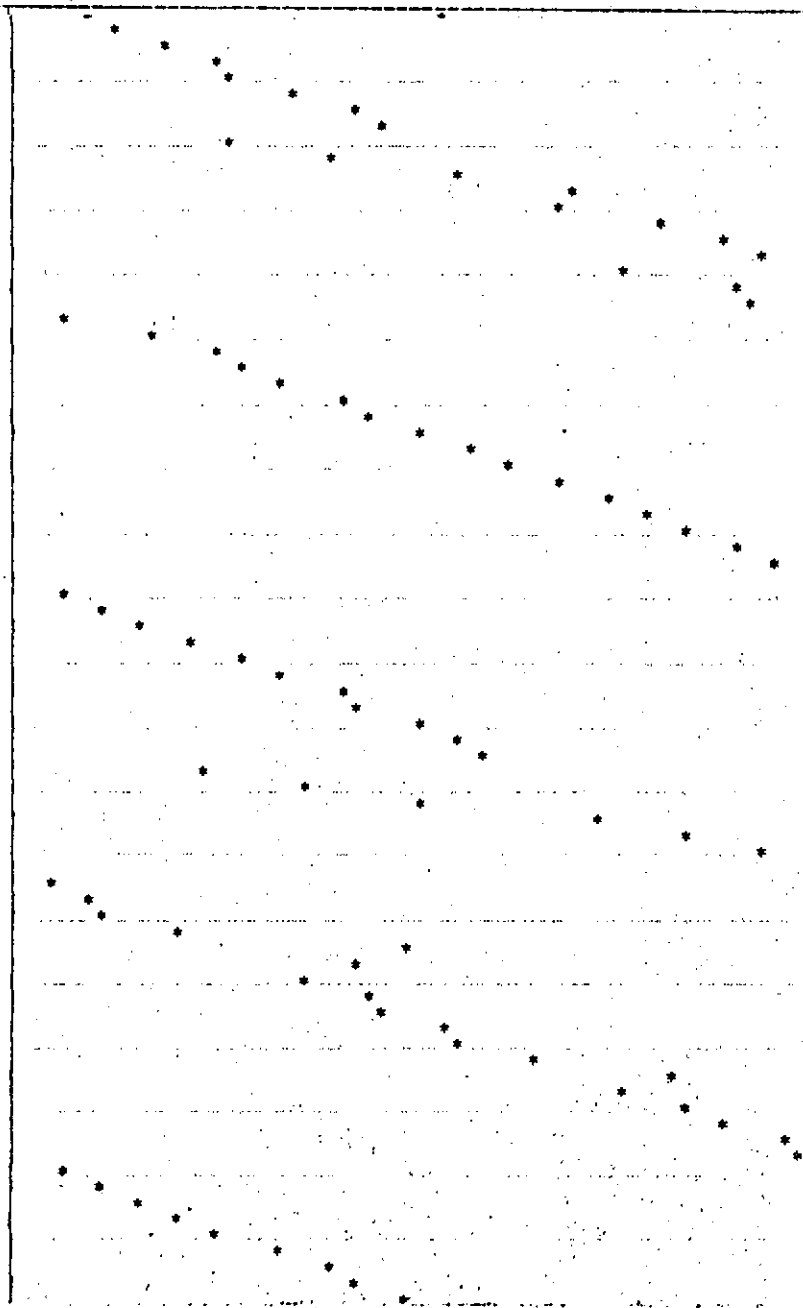
OMEGA PLOTS: C-E  
10 SECONDS PER LINE (DOWN); 1/64 LANE PER CHARACTER (ACROSS).

C+E

0

63 0

63



In addition, the program computes the mean, standard deviation, and spectral data for each Omega time slot, giving the number of data points at each 1/64 lane position. These statistics are computed for a "quick-look" at the entire data set. They are not useful directly for signal-statistical analysis. The control must be present. If it is blank, no plot is produced.

Note that all Omega station identifiers given on the program control card are referenced to 10.2 KHz. In order to operate the program with 13.6 KHz phase data, the user must remember that to plot the BD LOP at 13.6 KHz, he must specify the time slot (referenced to 10.2 KHz) in which the appropriate signals appear. In this case, his control card would read C-E to obtain the 13.6 BD LOP.

This program is written in standard FORTRAN IV for the IBM System/360 Model 44 at Ohio University. System-dependent code is avoided, so implementation on other computer systems should not be difficult.

C. Data Copy - DACOP. The program listing for DACOP is presented on the following page. DACOP, through use of direct-access files, allows the user to copy an 80-column card deck from one to nine times. The input deck must not contain more than 3,000 cards, or the program will copy only the first 3,000 input cards. A second DACOP run can then be made to copy remaining cards, if the deck is larger than 3,000 cards.

One control card is required, as the first input card. The format of the card is "COPY = N", where N is an integer from 1 to 9. If N is zero, one copy will be made. Note that this control card must precede the input deck. If it is missing, the first data card will be interpreted (wrongly) as the number of copies control card.

DACOP uses a FORTRAN data file which is written, rewound and read as if it were a sequential (tape) file. Direct-access placement of this sequential file is possible.

Output from DACOP consists of N copies of the input data deck, each copy followed by a blank card for separation. In addition, N copies of a printed listing of the data are produced, each separated by a page skip.

```

C...OMEGA PROGRAM FOR DECK COPIES 1-9
C...DACOP
C...
C...ONE CONTROL CARD MUST PRECEDE INPUT DECK
C...FORMAT IS "COPY=N" - N=1-9
C
C    INPUT DECK MAXIMUM IS 3,000 CARDS
C
C...OUTPUT IS N COPIES OF DECK, WITH BLANK CARD AFTER EACH.
C
C...R. W. LILLEY, AVIONICS, NOVEMBER, 1974
C
C    DIMENSION IN(20)
C    DEFINE FILE 4(3000,80,E,IDSK)
C    IC=1
C    PRINT 6
C...READ CONTROL CARD
C    READ(1,2)ICTL
C    2 FORMAT(5X,I1)
C...READ INPUT DECK
C    10 READ(1,1,END=20)IN
C    1    FORMAT(20A4)
C    1    IC=IC+1
C    1    IF(IC.GT.3000)GO TO 100
C    1    WRITE(4,1)IN
C    1    GO TO 10
C...LIST AND PUNCH OUTPUT DECKS
C    20    IC=IC-1
C    20    DO 31 K=1,ICTL
C    20    REWIND 4
C    20    DO 30 I=1,IC
C    20    READ(4,1)IN
C    20    PUNCH 1,IN
C    20    PRINT 7,IN
C    20    7 FORMAT(20X,20A4)
C    20    CONTINUE
C    20    PRINT 6
C    20    6 FORMAT('1')
C    20    PUNCH 1
C    20    31 CONTINUE
C    20    STOP
C    100 PRINT 101
C    101 FORMAT(' *** DECK TOO LARGE - FIRST 3,000 CARDS COPIED')
C    101 GO TO 20
C    101 END

```

### III. SUMMARY

The series of FORTRAN computer programs which is used for preparation of flight-test data for distribution to participants in the Tri-University Program in Air Transportation Systems and to other interested users has been described. Obviously, these programs are highly dependent upon data input formats from digital magnetic tape as written during flight tests. The future plans for data collection during test flights call for additional data to be recorded. We expect to multiplex flight path data (heading, airspeed and altitude) on the magnetic tape along with Omega phase measurements. Such plans, when implemented, may call for changes both in program input formats and in output data format.

It is noted that data output from the current reduction programs is in card form. With the additional data on flight path, it is expected that magnetic tape output for distribution will be required, due to the volume of data which will be recorded even on relatively short flights.

Card copies of any or all of the FORTRAN programs reported here may be obtained by contacting the author.

#### IV. ACKNOWLEDGEMENTS

The author wishes to acknowledge the valuable support of the Omega project team at Ohio University, and in particular Mr. Kent Chamberlin, for assistance in preparing the Omega data-reduction computer programs and this paper. Dr. Richard H. McFarland is the Director of the Avionics Engineering Center, and Mr. Ralph Burhans is the Project Engineer for the Tri-University Program.